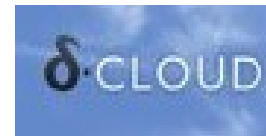




Supporting and Using EC2/CIMI on top of Cloud Environments via Deltacloud



Oved Ourfali
Senior Software Engineer – Red Hat

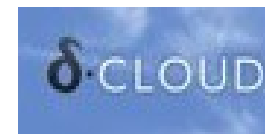
FOSDEM
February 2013

Agenda

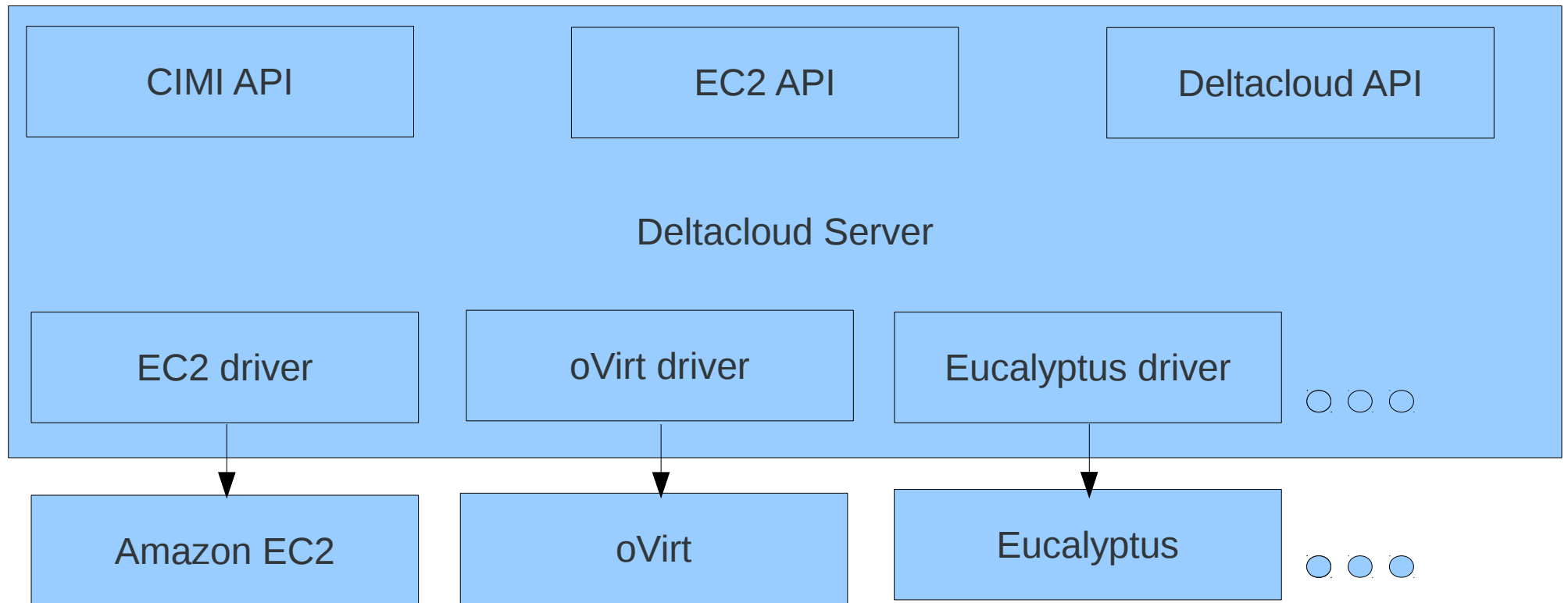
- ◆ Overview and Motivation
- ◆ Cloud APIs
- ◆ Using Deltacloud on top of oVirt via EC2 and CIMI
 - ◆ Integration
 - ◆ Installation
 - ◆ EC2/CIMI examples
- ◆ Future Work
- ◆ Summary

Deltacloud

- ◆ Open source Apache project
- ◆ Abstracts the differences between cloud providers
- ◆ Supports Deltacloud, EC2 and CIMI APIs
- ◆ Supports many cloud providers
 - ◆ EC2
 - ◆ oVirt
 - ◆ Eucalyptus
 - ◆ OpenNebula
 - ◆ OpenStack
 - ◆



Deltacloud



Motivation



- ♦ Heterogeneous cloud and virtualization environment
- ♦ Existing software working with common cloud APIs like EC2
 - ♦ Aeolus / CloudForms – cloud management software
 - ♦ Synaps – CloudWatch implementation over EC2 API
 - ♦ Heat (Openstack project)
 - ♦ CloudFormation and CloudWatch support
 - ♦ POC to support Deltacloud
 - ♦ Automated scripts
 - ♦

**If you work with
EC2 / CIMI / Deltacloud API
you can use it on top of many different
cloud providers!**

Cloud APIs



- ◆ DMTF CIMI API
 - ◆ Cloud Infrastructure Management Interface
 - ◆ <http://dmtf.org/standards/cloud>
 - ◆ Version 1.0.0 was published in August, 29th, 2012, 1.0.1 in October
 - ◆ Still new API, but aims to be the cloud standard API
- ◆ EC2 API – Amazon Elastic Cloud API
- ◆ Deltacloud API

CIMI API

- ◆ REST based API
- ◆ Main entities:
 - ◆ Machines - configuration, image, template
 - ◆ Volumes - images, configuration, image, template
 - ◆ Networks – Configuration, Forwarding Group, Template
 - ◆ Network Ports – Port configuration, Port Template

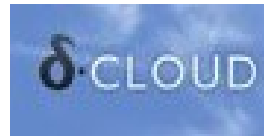
EC2 API



- ◆ API to Amazon Elastic Cloud, that allows to perform various actions
- ◆ HTTP GET/POST
 - ◆ Endpoint—Entry point to act on
 - ◆ Action—Action to perform on the endpoint
 - ◆ Parameters—Request parameters
- ◆ SOAP

- ◆ Main entities:
 - ◆ AMI – template that contains a software configuration
 - ◆ Instance Type – instance hardware configuration
 - ◆ Instance – compute resource
 - ◆ EBS Volume – storage resource
 - ◆ Networking entities (IP Addresses, NetworkInterface, Network ACL, Network ACL entries....)
 - ◆ Region – different geographical locations where data centers reside
 - ◆ Availability Zone – isolated from failures in other zones

Deltacloud API



- ♦ REST based API
- ♦ Main entities:
 - ♦ Instances
 - ♦ Images
 - ♦ Hardware Profiles
 - ♦ Realms
 - ♦ Storage Volumes and Snapshots
 - ♦ Networking Entities
 - ♦ IP Addresses
 - ♦ Load Balancers
 - ♦ Firewalls

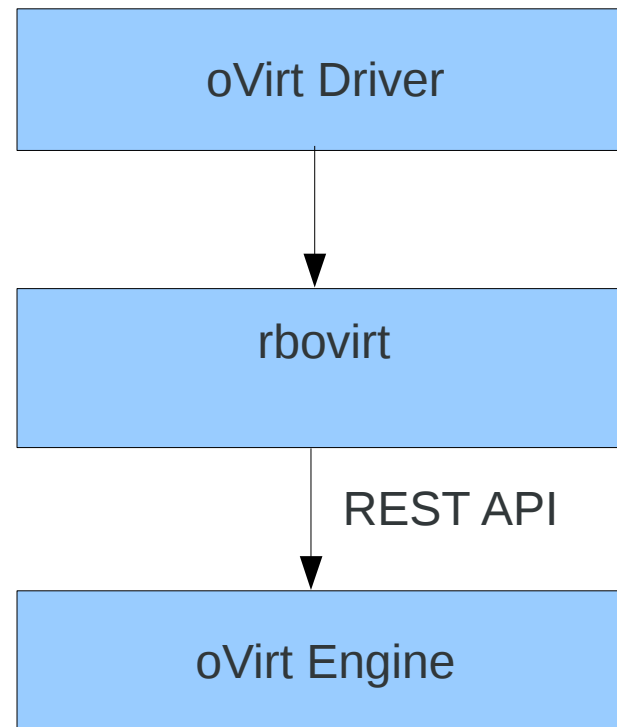
Using Deltacloud on top of oVirt via EC2 and CIMI

oVirt Engine



- ◆ The oVirt engine is a large scale, centralized virtualization management platform
- ◆ Provides IaaS capabilities
- ◆ Every capability is exposed via rich API, UI, CLI and SDK

Deltacloud-oVirt integration



Install and Run the Deltacloud Server

◆ Installing Deltacloud via GEM

Run the following commands as super user:

```
$ yum install rubygems gcc-c++ libxml libxslt libxslt-devel  
$ gem install rake deltacloud-core
```

◆ Running Deltacloud

- ◆ First set the API_PROVIDER in the following format
 - ◆ API-URL; DataCenter-ID ; USER/ADMIN (default to ADMIN)
- ◆ Examples
 - ◆ API_PROVIDER="http://engine-server:port/api;dc-id;ADMIN"
 - ◆ API_PROVIDER="http://engine-server:port/api;;USER"
- ◆ Then, run the deltacloud daemon

```
$ export API_PROVIDER="http://ovirt-server:port/api;;USER"  
$ deltacloud -i rhevm -f cimi,ec2,deltacloud
```

Install and Run the Deltacloud Server

◆ Installing Deltacloud via RPM

```
Run the following command as super user:  
$ yum install deltacloud-core-all
```

◆ Configure Deltacloud

- ◆ Set the DRIVER, API_PROVIDER and API_FRONTEND in /etc/sysconfig/deltacloud-core
 - ◆ API_PROVIDER="http://ovirt-server:port/api;;USER"
 - ◆ API_FRONTEND="cimi,ec2,deltacloud"
 - ◆ DRIVER="rhevm"

◆ Running the Deltacloud service

- ◆ `systemctl start deltacloud-core.service`

Install and Run the Deltacloud Server

- ◆ Admin API requests are DC-specific, whereas User ones are system-wide
- ◆ The API_PROVIDER can be changed by setting the “X-Deltacloud-Provider” HTTP Header on every request

Basic Operations - EC2

- ◆ The following EC2 operations are supported when using Deltacloud on top of oVirt engine:
 - ◆ Get Clusters (Availability Zones)
 - ◆ Get Templates (AMI – Amazon Machine Images + Instance Type)
 - ◆ Get VMs (Instances)
 - ◆ Create new VM (Run Instances)
 - ◆ Start VM (Start Instances)
 - ◆ Stop VM (Stop Instances)
 - ◆ Delete VM (Terminate Instances)

Basic Operations - EC2

◆ Get oVirt Clusters

◆ HTTP GET

◆ <http://localhost:3001/ec2/?Action=DescribeAvailabilityZones>

```
<DescribeAvailabilityZonesResponse xmlns='http://ec2.amazonaws.com/doc/2012-04-01/'>
  <requestId>7c14876387aa8d47b2ff1b1edf0d253f</requestId>
  <availabilityZoneInfo>
    <item>
      <zoneName>99408929-82cf-4dc7-a532-9d998063fa95</zoneName>
      <zoneState>AVAILABLE</zoneState>
      <regionName>Default</regionName>
    </item>
    <item>
      <zoneName>77d52ef6-11e3-11e2-be8c-87a6485627fe</zoneName>
      <zoneState>AVAILABLE</zoneState>
      <regionName>barcelona</regionName>
    </item>
  </availabilityZoneInfo>
</DescribeAvailabilityZonesResponse>
```

Basic Operations - EC2

- ◆ Get oVirt Templates
 - ◆ HTTP GET
 - ◆ <http://localhost:3001/ec2/?Action=DescribeImages>

```
<DescribeImagesResponse xmlns='http://ec2.amazonaws.com/doc/2012-04-01/'>
  <requestId>a9f98b46f42bd96da30fe5b095617e0d</requestId>
  <imagesSet>
    <item>
      <imageId>000000000-0000-0000-0000-000000000000</imageId>
      <imageState>ok</imageState>
      <imageOwnerId>admin@internal</imageOwnerId>
      <architecture>x86_64</architecture>
      <imageType>machine</imageType>
      <name>Blank</name>
      <description>Blank template</description>
    </item>
    <item>
      <imageId>77ea7521-3017-4ee9-bcdf-d566178991c6</imageId>
      <imageState>ok</imageState>
      <imageOwnerId>admin@internal</imageOwnerId>
      <architecture>x86_64</architecture>
      <imageType>machine</imageType>
      <name>desktop_template</name>
      <description></description>
    </item>
  </imagesSet>
</DescribeImagesResponse>
```

Basic Operations - EC2

- ♦ Create VM from Template, in a Cluster

(EC2 - run an instance, created from an image, in an availability zone)

- ♦ HTTP GET

- ♦ `http://localhost:3001/ec2/?`

`Action=RunInstances&Placement.AvailabilityZone=77d52ef6-11e3-11e2-be8c-87a6485627fe&ImageId=77ea7521-3017-4ee9-bcdf-d566178991c6`

- ♦ Other options

- ♦ `UserData=.....` (Pass user data to the VM via hook. Will need to support the native oVirt VM Payload feature)
- ♦ `InstanceType=...` (Will be supported when we introduce flavors in oVirt)

Basic Operations - EC2



◆ Response

```
<RunInstancesResponse xmlns='http://ec2.amazonaws.com/doc/2012-04-01/'>
  <requestId>daad0aaa272ee19e2c8645e77b620e38</requestId>
  <reservationId>r-11111111</reservationId>
  <ownerId>deltacloud</ownerId>
  <groupSet>
    <item>
      <groupId>sg-11111111</groupId>
      <groupName>default</groupName>
    </item>
  </groupSet>
  <instanceSet>
    <item>
      <instanceId>84623036-49fe-4f22-ac6f-ff526e8b82be</instanceId>
      <imageId>77ea7521-3017-4ee9-bcdf-d566178991c6</imageId>
      <instanceType>SERVER</instanceType>
      <launchTime>2012-10-10T09:53:08.984+02:00</launchTime>
      <ipAddress>192.168.1.27</ipAddress>
      <dnsName>192.168.1.27</dnsName>
      <architecture>i386</architecture>
      <instanceState>
        <code>0</code>
        <name>pending</name>
      </instanceState>
      <placement>
        <availabilityZone>77d52ef6-11e3-11e2-be8c-87a6485627fe</availabilityZone>
        <groupName></groupName>
        <tenancy>default</tenancy>
      </placement>
    </item>
  </instanceSet>
</RunInstancesResponse>
```

Basic Operations - EC2

◆ Get VM

- ◆ HTTP GET
- ◆ All instances - `http://localhost:3001/ec2/?Action=DescribeInstances`
- ◆ Specific instance - `http://localhost:3001/ec2/?Action=DescribeInstances&InstanceId.1=84623036-49fe-4f22-ac6f-ff526e8b82be`

```
<DescribeInstancesResponse xmlns='http://ec2.amazonaws.com/doc/2012-04-01/'>
  <instanceSet>
    <item>
      <instanceId>84623036-49fe-4f22-ac6f-ff526e8b82be</instanceId>
      <imageId>77ea7521-3017-4ee9-bcdf-d566178991c6</imageId>
      <instanceType>SERVER</instanceType>
      <launchTime>2012-10-10T09:53:08.984+02:00</launchTime>
      <ipAddress>192.168.1.27</ipAddress>
      <dnsName>192.168.1.27</dnsName>
      <architecture>i386</architecture>
      <instanceState>
        <code>80</code>
        <name>stopped</name>
      </instanceState>
      <placement>
        <availabilityZone>77d52ef6-11e3-11e2-be8c-87a6485627fe</availabilityZone>
        <groupName></groupName>
        <tenancy>default</tenancy>
      </placement>
    </item>
  </instanceSet>
</DescribeInstancesResponse>
```

Basic Operations - EC2

Start VM

HTTP GET

<http://localhost:3001/ec2/?>

Action=**StartInstances**&InstanceId.1=84623036-49fe-4f22-ac6f-ff526e8b82be”

```
<StartInstancesResponse xmlns='http://ec2.amazonaws.com/doc/2012-04-01/'>
  <requestId>0abf89a0b59c9f090818872f574766e5</requestId>
  <instancesSet>
    <item>
      <instanceId>84623036-49fe-4f22-ac6f-ff526e8b82be</instanceId>
      <currentState>
        <code>0</code>
        <name>pending</name>
      </currentState>
      <previousState>
        <code>-1</code>
        <name>unknown</name>
      </previousState>
    </item>
  </instancesSet>
</StartInstancesResponse>
```


Basic Operations - EC2

◆ Stop VM

◆ HTTP GET

◆ <http://localhost:3001/ec2/?>

Action=**StopInstances**&InstanceId.1=84623036-49fe-4f22-ac6f-ff526e8b82be”

```
<StopInstancesResponse xmlns='http://ec2.amazonaws.com/doc/2012-04-01/'>
  <requestId>49969b2552340028a4bda5a09e536778</requestId>
  <instancesSet>
    <item>
      <instanceId>84623036-49fe-4f22-ac6f-ff526e8b82be</instanceId>
      <currentState>
        <code>64</code>
        <name>stopping</name>
      </currentState>
      <previousState>
        <code>-1</code>
        <name>unknown</name>
      </previousState>
    </item>
  </instancesSet>
</StopInstancesResponse>
```

Basic Operations - EC2

◆ Delete VM

◆ HTTP GET

◆ <http://localhost:3001/ec2/?>

Action=**TerminateInstances**&InstanceId.1=84623036-49fe-4f22-ac6f-ff526e8b82be”

```
<TerminateInstancesResponse xmlns='http://ec2.amazonaws.com/doc/2012-04-01/'>
  <requestId>67392f70be9b6759ba830b85153661f5</requestId>
  <instancesSet>
    <item>
      <instanceId>5ee314f5-a01b-41f3-81f7-fe4acb6c31f9</instanceId>
      <currentState>
        <code>32</code>
        <name>shutting-down</name>
      </currentState>
      <previousState>
        <code>-1</code>
        <name>unknown</name>
      </previousState>
    </item>
  </instancesSet>
</TerminateInstancesResponse>
```

Basic Operations - CIMI

- ◆ The following CIMI operations are supported when using Deltacloud on top of oVirt engine:
 - ◆ Get Templates (Get Images)
 - ◆ Get VMs (Get Machines)
 - ◆ Create new VM (Create Machine)
 - ◆ Start VM (Start Machine)
 - ◆ Stop VM (Stop Machine)
 - ◆ Delete VM (Delete Machine)

Basic Operations - CIMI

◆ Get Templates

- ◆ HTTP GET
- ◆ http://localhost:3001/cimi/machine_images

```
<Collection xmlns="http://schemas.dmtf.org/cimi/1">
  <id>http://localhost:3001/cimi/machine_images</id>
  <name>default</name>
  <description>MachineImage Collection for the Rhevm driver</description>
  <count>3</count>
  <MachineImage>
    <id>http://localhost:3001/cimi/machine_images/00000000-0000-0000-0000-000000000000</id>
    <name>00000000-0000-0000-0000-000000000000</name>
    <description>Blank template</description>
    <created>2012-10-11 11:20:26 +0200</created>
    <imageLocation href="rhevm://00000000-0000-0000-0000-000000000000" />
  </MachineImage>
  <MachineImage>
    <id>http://localhost:3001/cimi/machine_images/77ea7521-3017-4ee9-bcdf-d566178991c6</id>
    <name>77ea7521-3017-4ee9-bcdf-d566178991c6</name>
    <description>Desktop template</description>
    <created>2012-10-11 11:20:26 +0200</created>
    <imageLocation href="rhevm://77ea7521-3017-4ee9-bcdf-d566178991c6" />
  </MachineImage>
  <MachineImage>
    <id>http://localhost:3001/cimi/machine_images/a10f0123-ca40-4428-adac-26ac42f282dd</id>
    <name>a10f0123-ca40-4428-adac-26ac42f282dd</name>
    <description>Workshop template</description>
    <created>2012-10-11 11:20:26 +0200</created>
    <imageLocation href="rhevm://a10f0123-ca40-4428-adac-26ac42f282dd" />
  </MachineImage>
</Collection>
```

Basic Operations - CIMI

- ◆ Create VM
 - ◆ HTTP POST
 - ◆ <http://localhost:3001/cimi/machines>
 - ◆ Body

```
<Machine>
  <name>cimi_machine</name>
  <description>My first machine!</description>
  <machineTemplate>
    <machineConfig
      href="http://localhost:3001/cimi/machine_configurations/SERVER"/>
    <machineImage
      href="http://localhost:3001/cimi/machine_images/a10f0123-ca40-4428-adac-26ac42f282dd"/>
    </machineTemplate>
  </Machine>
```

Basic Operations - CIMI



➤ Response:

```
<Machine xmlns="http://schemas.dmtf.org/cimi/1">
  <id>http://localhost:3001/cimi/machines/7434e6e3-a85d-4bfb-b952-b191288c2aa2</id>
  <name>7434e6e3-a85d-4bfb-b952-b191288c2aa2</name>
  <description>cimi_machine</description>
  <created>2012-10-18T14:22:57.675+02:00</created>
  <property name="machine_image">http://localhost:3001/cimi/machine_images/a10f0123-ca40-4428-adac-26ac42f282dd</property>
  <property name="credential">http://localhost:3001/cimi/credentials</property>
  <state>PENDING</state>
  <cpu>2</cpu>
  <memory>1048576</memory>
  <disks href="http://localhost:3001/cimi/machines/7434e6e3-a85d-4bfb-b952-b191288c2aa2/disks">
    <id>http://localhost:3001/cimi/machines/7434e6e3-a85d-4bfb-b952-b191288c2aa2/disks</id>
    <count>1</count>
    <Disk>
      <id>http://localhost:3001/cimi/machines/7434e6e3-a85d-4bfb-b952-b191288c2aa2/disks/
7434e6e3-a85d-4bfb-b952-b191288c2aa2_disk_1024</id>
      <name>7434e6e3-a85d-4bfb-b952-b191288c2aa2_disk_1024</name>
      <description>DiskCollection for Machine 7434e6e3-a85d-4bfb-b952-b191288c2aa2</description>
      <created>2012-10-18T14:22:57.675+02:00</created>
      <capacity>1024</capacity>
    </Disk>
  </disks>
  <volumes href="http://localhost:3001/cimi/machines/7434e6e3-a85d-4bfb-b952-b191288c2aa2/volumes" />
</Machine>
```

Basic Operations - CIMI

➤ Get VM

➤ HTTP GET

- <http://localhost:3001/cimi/machines/7434e6e3-a85d-4bfb-b952-b191288c2aa2>

```
<Machine xmlns="http://schemas.dmtf.org/cimi/1">
  <id>http://localhost:3001/cimi/machines/7434e6e3-a85d-4bfb-b952-b191288c2aa2</id>
  <name>7434e6e3-a85d-4bfb-b952-b191288c2aa2</name>
  <description>cimi_machine</description>
  <created>2012-10-18T14:22:57.675+02:00</created>
  <property name="machine_image">
    http://localhost:3001/cimi/machine_images/a10f0123-ca40-4428-adac-26ac42f282dd
  </property>
  <property name="credential">http://localhost:3001/cimi/credentials</property>
  <state>STOPPED</state>
  <cpu>2</cpu>
  <memory>1048576</memory>
  <disks href="http://localhost:3001/cimi/machines/7434e6e3-a85d-4bfb-b952-b191288c2aa2/disks">
    ....
  </disks>
  <volumes href="http://localhost:3001/cimi/machines/7434e6e3-a85d-4bfb-b952-b191288c2aa2/volumes" />
  <operation rel="http://schemas.dmtf.org/cimi/1/action/start"
    href="http://localhost:3001/cimi/machines/7434e6e3-a85d-4bfb-b952-b191288c2aa2/start" />
  <operation rel="http://schemas.dmtf.org/cimi/1/action/destroy"
    href="http://localhost:3001/cimi/machines/7434e6e3-a85d-4bfb-b952-b191288c2aa2" />
</Machine>
```

Basic Operations - CIMI

- Start VM

- HTTP POST

- <http://localhost:3001/cimi/machines/81e0908a-04da-4859-8a92-f80d071a2e4c/start>

- Body:

```
<Action>  
  <action>http://www.dmtf.org/cimi/action/start</action>  
</Action>
```

- Response:

- HTTP CODE 200

Basic Operations - CIMI

◆ Stop VM

◆ HTTP POST

◆ <http://localhost:3001/cimi/machines/81e0908a-04da-4859-8a92-f80d071a2e4c/stop>

◆ Body:

```
<Action>  
  <action>http://www.dmtf.org/cimi/action/stop</action>  
</Action>
```

◆ Response:

◆ HTTP CODE 200

Basic Operations - CIMI

- ◆ Delete VM

- ◆ HTTP DELETE

- ◆ `http://localhost:3001/cimi/machines/1bac66da-7ecc-4e19-bc91-0623c5448fc9`

- ◆ Response:

- ◆ HTTP CODE 200

Future Work (oVirt / Deltacloud)

◆ General

- ◆ Supporting VM configuration/instance type (not yet part of oVirt)
- ◆ Better mapping and support for networks
- ◆ Better mapping and support for storage volumes (Integrate into Deltacloud)
- ◆ Supporting VM Payload (Deltacloud)

◆ EC2

- ◆ Supporting more EC2 options/actions (oVirt / Deltacloud)

◆ CIMI

- ◆ Supporting more CIMI options/actions/entities (oVirt / Deltacloud)
- ◆ Support updating resources (Deltacloud will need to support that and integrate with oVirt)

Summary



- ◆ Enabling the use of common cloud APIs on top of cloud environments is very useful
- ◆ Deltacloud exposes common cloud APIs on top of many cloud providers, including oVirt
- ◆ Projects who would like to use the Deltacloud-oVirt integration - we would be happy to know about it, guide you, and help! Please contact us on users@ovirt.org

Useful Links



- ◆ Deltacloud
 - ◆ Developer mailing list dev@deltacloud.apache.org
 - ◆ <http://deltacloud.apache.org>
 - ◆ #deltacloud on Freenode
- ◆ oVirt
 - ◆ Users mailing list users@ovirt.org
 - ◆ <http://ovirt.org>
- ◆ My blog
 - ◆ <http://ovedou.blogspot.com>

oVirt

Questions





THANK YOU !

<http://www.ovirt.org>

ovedo@redhat.com
ovedo at #ovirt (irc.oftc.net)